# Лабораторная работа №7 "Функции в С++"

**Вариант №15**

**Цель:** 1) Получить практические навыки работы с функциями;

2) получить практические навыки работы с шаблонами функций;

3) получить практические навыки работы с указателями функций.

**Задача:** 1. Написать функцию с умалчиваемыми параметрами в соответствии с вариантом, продемонстрировать различные способы вызова функции:

* с параметрами заданными явно,
* с опущенными параметрами
* часть параметров задана явно, а часть опущена.

2. Написать функцию с переменным числом параметров в соответствии с вариантом, продемонстрировать вызов функции с различным числом параметров.

3. Написать перегруженные функции в соответствии с вариантом. Написать демонстрационную программу для вызова этих функций.

4. Написать шаблон функций вместо перегруженных функций из задания 3. Написать демонстрационную программу для вызова этих функций. списка параметров

5. Решить уравнение указанным в варианте методом. Уравнение передать в функцию как параметр с помощью указателя.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 15 | Печать фамилии, имени и рейтинга | Максимальный из элементов в списке параметров, стоящих на четных местах | Среднее арифметическое массива | Метод половинного деления  3x - 4lnx - 5 = 0  Отрезок, содержащий корень: [2;4]  Точное значение: 3,2300 |

**Решение:**

#include <iostream>

#include <string>

#include <math.h>

#include <iomanip>

using namespace std;

// 1. Умалчиваемые параметры

int Card(string surname = "Сташков", string name = "Арсений", string rating = "10")

{

cout << surname << " " << name << " " << rating << "\n";

return 0;

}

// 2. Переменное число параметров

int Options(int a, ...)

{

int max = 0;

int res;

for (int\* p = &a; a >= 0; a--)

{

res = \*(p++);

if (res > max && a%2 == 0)

{

max = res;

}

}

cout << max << "\n";

return 0;

}

// 3. Перегрузка функций (int)

int Array(int a[5])

{

int average = 0;

for (int i = 0; i < 5; i++)

{

average += a[i];

}

return average / 5;

}

// Перегрузка функций (double)

double Array(double a[5])

{

double average = 0;

for (int i = 0; i < 5; i++)

{

average += a[i];

}

return average / 5;

}

// Перегрузка функций (char)

char Array(char a[5])

{

double average = 0;

for (int i = 0; i < 5; i++)

{

average += (int)a[i];

}

return average / 5;

}

// Шаблон функции (int)

template <class T>

T ArrayTemplate1(T a[5])

{

int average = 0;

for (int i = 0; i < 5; i++)

{

average += a[i];

}

return average / 5;

}

// Шаблон функции (double)

template <class T>

T ArrayTemplate2(T a[5])

{

double average = 0;

for (int i = 0; i < 5; i++)

{

average += a[i];

}

return average / 5;

}

// Шаблон функции (char)

template <class T>

T ArrayTemplate3(T a[5])

{

double average = 0;

for (int i = 0; i < 5; i++)

{

average += (int)a[i];

}

return average / 5;

}

// 4. Метод половинного деления (функция Уравнения)

double Equation(double x)

{

return 3 \* x - 4 \* log(x) - 5;

}

// Метод половинного деления (функция Решения)

double HalfDevisionMethod(double a, double b, double (\*Equation)(double))

{

double c, result;

while (b - a > 1e-2) {

c = (a + b) / 2;

if (Equation(b) \* Equation(c) < 0)

{

a = c;

}

else

{

b = c;

}

}

result = (a + b) / 2;

cout << " " << fixed << setprecision(4) << "3x - 4lnx - 5 = 0; x = " << result << "\n";

if (result > 3.2300)

{

cout << "\nРезультат " << fixed << setprecision(4) << result << " больше, чем 3.2300\n";

}

else

{

cout << "\nРезультат " << fixed << setprecision(4) << result << " меньше, чем 3.2300\n";

}

return 0;

}

int main()

{

setlocale(LC\_ALL, "Rus");

cout << "Функция с умалчиваемыми параметрами:\n\n";

// Параметры опущены

Card();

// Параметры указаны явно

Card("Прохоров", "Дмитрий", "12");

// Часть параметров указана явно

Card("Савельев", "Михаил");

cout << "\nФункция с переменным числом параметров:\n\n";

// Задаем параметры

Options(5, 20, 6, 15, 200, 100);

cout << "\nПерегруженные функции и шаблон функции:\n\n";

// Вызываем функцию с типом int

int arr1[] = { 0, 1, 2, 3, 8 };

cout << "int\t" << Array(arr1) << "\n\n";

// Вызываем функцию с типом double

double arr2[] = { 0, 1, 2, 3, 8 };

cout << "double\t" << Array(arr2) << "\n\n";

// Вызываем функцию с типом char

char arr3[] = { '0', '1', '2', '9', '9' };

cout << "char\t" << Array(arr3) << "\n\n";

// Вызываем функцию с шаблонным типом int

int arr4[] = { 6, 10, 2, 3, 8 };

cout << "Шаблон int\t" << ArrayTemplate1(arr4) << "\n\n";

// Вызываем функцию с шаблонным типом double

double arr5[] = { 6, 1.5, 2.5, 3, 8.8 };

cout << "Шаблон double\t" << ArrayTemplate2(arr5) << "\n\n";

// Вызываем функцию с шаблонным типом char

char arr6[] = { '6', '1', '2', '3', '5' };

cout << "Шаблон char\t" << ArrayTemplate3(arr6) << "\n\n";

cout << "\nПередача функции как параметра другой функции с помощью указателя:\n\n";

// Передаем указатель на функцию Уравнения функции Решения

HalfDevisionMethod(2, 4, \*Equation);

return 0;

}

**Результат:**

Функция с умалчиваемыми параметрами:

Сташков Арсений 10

Прохоров Дмитрий 12

Савельев Михаил 10

Функция с переменным числом параметров:

100

Перегруженные функции и шаблон функции:

int 2

double 2.8

char 4

Шаблон int 5

Шаблон double 4.36

Шаблон char 3

Передача функции как параметра другой функции с помощью указателя:

3x - 4lnx - 5 = 0; x = 3.2305

Результат 3.2305 больше, чем 3.2300